Intelligent Neurosurgical Patient Profiling Model

Project Proposal

Supervisor: Peter X.Liu

Team members:

Bingtao Liu 101029143

Xiang Cheng 101085518

Defa Hu 101038610

Jiayi Chen 101075620

Zejian Xing 101082830

October 18th, 2021

Content

1.0 Introduction

The purpose of this project is to create models which are equivalent or better than the physician's current understanding of predicted particular outcomes, for example, the type of brain tumour or clinical outcomes (survival rates, surgery necessity, etc.). The results would be published in a medical/clinical journal and possibly computer science/machine learning journals. If successful, the models could be further developed for wider applications or use in clinical practice. The project will involve training a neural network using CT/MRI scans of neurosurgical patients, which will essentially be series of 2D black/white images, in addition to clinical information (age, sex, and more specific factors relating to their disease).

The team of the project is consisting of five students in Carleton University's fourth-year Engineering program. One student of the team, Jiayi Chen is in Biomedical and Electrical Engineering program. Two of the team, Xiang Cheng and Zejian Xing are Electrical Engineering majors. And the other two students, Defa Hu and Bingtao Liu are in Software Engineering and Computer Systems Engineering.

2.0 Challenges of the project

The main challenge of this project would be the accuracy of the outcome. The outcome is dependant on the model training process. A training model is a dataset that is used to train a machine learning (ML) algorithm. Such dataset consists of sample input and corresponding output data and such data will be used to train the algorithm to correlate the desired way. The model will then be modified as a result of this. [1]

Since model training is a process of feeding the algorithm with data to help identify and learn good values for all attributes involved. To get a functional and accurate model, mass data and time will be needed.

3.0 Project foundation

3.1 Background of the project

The technology of facial recognition in our smartphones, autonomous modes in self-driving cars, and diagnostic imaging in healthcare have made massive strides in recent years. They all use solutions that make sense of objects in front of them - hence why it's often called "computer vision". These computers are able to make accurate decisions based on what they "see".[2] With the continuous development and progress of medical imaging technology and computer technology, medical image analysis has become an indispensable tool and technical means in medical research, clinical disease diagnosis and treatment. In recent years, Deep Learning especially Deep Convolutional neural networks (CNNs), has rapidly developed into a research hotspot of medical image analysis, which can automatically extract disease diagnosis features hidden in medical image big data. Therefore, when a patient has completed an MRI or CT scan, if there is a software that can make a preliminary diagnosis in advance, and then inform the attending doctor of the diagnosis result, this will effectively reduce the time the attending doctor spends on diagnosing the disease. And when encountering a rare case, if the technician has entered the data of the rare case in advance, then this software will provide great convenience to the medical staff.

3.2 Measurable functional and non-functional requirements

One of the major measurable functional requirements is that the trained model would predict a particular medical outcome. With the input of CT/MRI of a patient along with their clinical information, the model is supposed to determine if the patient can be diagnosed with certain type of neurosurgical disease, or if certain type of medical operation needs to be performed. With known cases gathered from internet resource as benchmark, the model is to be tested along with the development of machine learning process to evaluate the accuracy.

One of the non-functional requirements is to be equipped with clear, responsive and intuitive user interface. For maintainability of the project, we will have a test set to make sure the code is running properly and update it through the project. For the data we used to train the ML algorithm, we have to make sure they are correct and secure.

4.0 Project construction

4.1 Required skills

Image recognition: The ability of software to identify objects, places, people, writing and actions in images. In this project, it will be based on Machine Learning Deep Learning. Computer can use machine vision technologies in combination with a camera and artificial intelligence software to achieve image recognition.

Machine Learning: An application of artificial intelligence (AI) that provides systems the ability to automatically learn and improve from experience without being explicitly programmed.

Python: A general-purpose programming language, used for implementing machine learning deep learning algorithms in this project.

TensorFlow: It is an open-source artificial intelligence library, using data flow graphs to build models. It allows developers to create large-scale neural networks with many layers. TensorFlow is mainly used for: Classification, Perception, Understanding, Discovering, Prediction and Creation.

Jupyter Notebook: “An interactive computational environment, work with the Python language and data.” It includes inputs and outputs of Machine Learning, images, mathematics, etc. [3]

4.2 Area of focus for individual member

Member	Bingtao Liu	Zejian Xing	Xiang Cheng	Defa Hu	Jiayi Chen
Category	Machine Learning	Image Information	Software Feature selection	Machine Learning	Machine Learning
Area of focus	Train Model	Data Analysis	Python code integration	Build Model	Test/Validate Model

4.3 Description of the methods

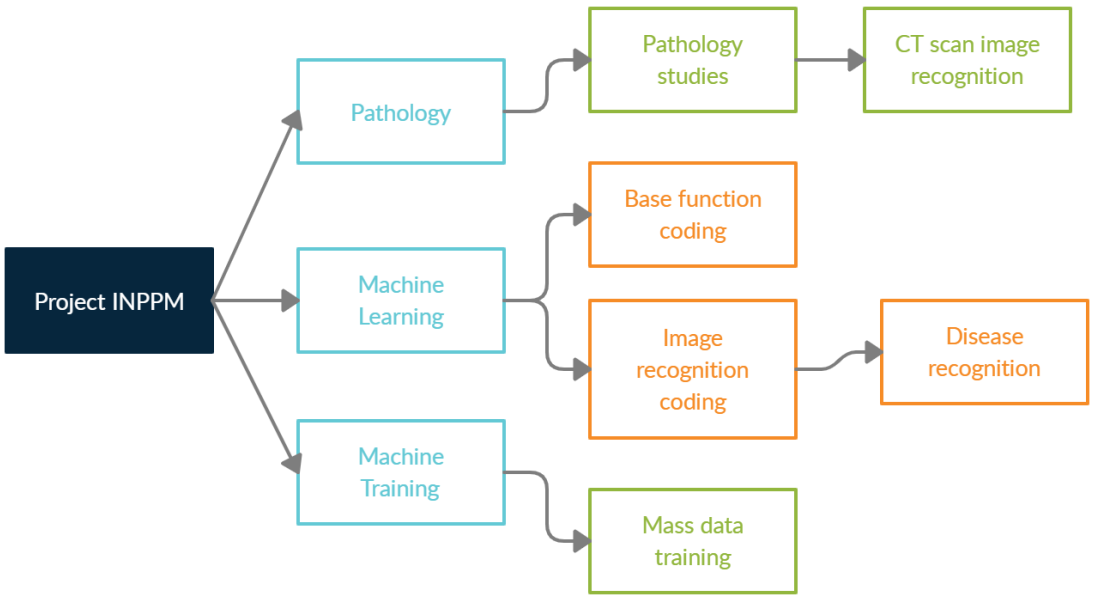

Image.1

The project will be constructed with 3 major components, pathology, machine learning, and machine training. The pathology will include pathology studies and CT scan image recognition for disease characteristics. Then the machine learning construction will take place. This includes the basic functionality coding, along with the image recognition to

be coded into the program for the machine to study the disease characteristics. Lastly, the machine training will take place where mass data will be injected for the machine to study and analyze for accurate results.

The first step of the project is pathology.

We can use the function tensorflow.io.read_file to take the file name as the required argument, then return the content of the file by tensorflow.string. For example, we need to input the soccer ball image.[4]

```
import tensorflow

values = tf.io.read_file('soccer_ball.jpg')
```

Image.2 [4]

Load the image

```
def decode_image(filename, image_type, resize_shape, channels):
    value = tensorflow.io.read_file(filename)
    if image_type == 'png':
        decoded_image = tensorflow.image.decode_png(value,
channels=channels)
    elif image_type == 'jpeg':
        decoded_image = tensorflow.image.decode_jpeg(value,
channels=channels)
    else:
        decoded_image = tensorflow.image.decode_image(value,
channels=channels)

    if resize_shape is not None and image_type in ['png', 'jpeg']:
        decoded_image = tf.image.resize(decoded_image, resize_shape)

    return decoded_image
```

Image.3 [4]

Deal with image data and store them.

```
import tensorflow as tf


def get_dataset(image_paths, image_type, resize_shape, channels):
    filename_tensor = tf.constant(image_paths)
    dataset = tf.data.Dataset.from_tensor_slices(filename_tensor)

    def _map_fn(filename):
        decode_images = decode_image(filename, image_type,
resize_shape, channels=channels)
        return decode_images

    map_dataset = dataset.map(_map_fn) # we use the map method: allow
to apply the function _map_fn to all the
    # elements of dataset
    return map_dataset
```

Image.4 [4]

One way we can extract the decode image data from dataset is using tensorflow.data.Iterator. Use the get_next function to obtain a next_element tensor.

```
def get_image_data(image_paths, image_type, resize_shape, channels):
    dataset = get_dataset(image_paths, image_type, resize_shape,
channels)
    iterator = tf.compat.v1.data.make_one_shot_iterator(dataset)
    next_image = iterator.get_next()

    return next_image
```

Image.5 [4]

Then is the second step, we will build the model for the project.

There is a sample of creating the model of a project. The Sequential model consists of three convolution blocks with a max pooling layer. And there is a fully connected layer with 128 bits on top.

```
num_classes = 5

model = Sequential([
  layers.Rescaling(1./255, input_shape=(img_height, img_width, 3)),
  layers.Conv2D(16, 3, padding='same', activation='relu'),
  layers.MaxPooling2D(),
  layers.Conv2D(32, 3, padding='same', activation='relu'),
  layers.MaxPooling2D(),
  layers.Conv2D(64, 3, padding='same', activation='relu'),
  layers.MaxPooling2D(),
  layers.Flatten(),
  layers.Dense(128, activation='relu'),
  layers.Dense(num_classes)
])
```

Image.6 [5]

Then compile the model of the sample. The tf.keras.optimizers.Adam optimizer and tf.keras.losses.SparseCategoricalCrossentropy loss function are used in this step.

```
model.compile(optimizer='adam',
              loss=tf.keras.losses.SparseCategoricalCrossentropy(from_logits=True),
              metrics=['accuracy'])
```

Image.7 [5]

After that, we can summary the model by Model.summary method.

```
model.summary()
```

Image.8 [5]

Next step, we will train and test the model for the project.

From the sample below, the function model.fit will be used in the training.

```
epochs=10
history = model.fit(
  train_ds,
  validation_data=val_ds,
  epochs=epochs
)
```

Image.9 [5]

After the training function set up, we will visualize training results. The plots of loss and accuracy on the training will be created.

```
acc = history.history['accuracy']
val_acc = history.history['val_accuracy']

loss = history.history['loss']
val_loss = history.history['val_loss']

epochs_range = range(epochs)

plt.figure(figsize=(8, 8))
plt.subplot(1, 2, 1)
plt.plot(epochs_range, acc, label='Training Accuracy')
plt.plot(epochs_range, val_acc, label='Validation Accuracy')
plt.legend(loc='lower right')
plt.title('Training and Validation Accuracy')

plt.subplot(1, 2, 2)
plt.plot(epochs_range, loss, label='Training Loss')
plt.plot(epochs_range, val_loss, label='Validation Loss')
plt.legend(loc='upper right')
plt.title('Training and Validation Loss')
plt.show()
```

Image.10 [5]

After it, we will get the accuracy on the validation set. By the accuracy, we can do the data augmentation, recompile and retrain the model to get the better result.

4.4 project risks

For this project, except for the only student majoring in biomedicine and electronics in this group, the other team members are the first time to get in touch with biomedicine in this project, and all members need to learn new things about this project. knowledge.

What we plan is that when the patient's MRI or CT image is scanned to the computer, the software in the computer can convert the image into a computer readable data file, and then view different sections after reading the code, and automatically find the lesion through data comparison, and diagnose.

One of the risks of this project is the low accuracy of diagnosis in the early stage. Since we will use machine learning, most of the previous results will have no reference or be unavailable before there is enough data. Therefore, we need to cooperate with multiple parties to obtain more information to support our project. Data of rare cases is more preferred and will improve the professionalism of our software.

The second risk is that python is a new language that were not taught from the courses. Even though it is the preferred language for machine learning and easier to operate, it is a challenge for all of the group members to start from scratch.

The risk of coding with a completely new language is the possibility of bugs which will cause malfunctions during software operation. Therefore, the software must undergo extreme testing before releasing and promoting the software, so that it can be modified in time when an abnormality occurs. Automatic testbenches and monitoring may be designed and implemented so that the operator can be notified in time when a fault occurs.

The machine learning algorithm’s original choices might not fit the actual case to perform a reasonable prediction. The mitigation is choosing different machine learning algorithms that might fit the actual case before we start to implement the actual machine learning algorithm.

5.0 Project relation

In the group, there are two members that currently studies Electrical Engineering. The project will be based on machine learning which involve microprocessor and software programming. The microprocessor programming may be in future standalone product where the machine will include many microprocessors in which the system can operate fully by its own without the need for a PC. The software programming involved will be AI based, which is also a major part of the electrical engineering. It is important to study AI based programming as in the future career there will only be more autonomous engineering involved, and such skill will be required while an engineer is designing circuit boards and other products.

The group member who studies Biomedical and Electrical Engineering has a similar concept as Electrical Engineering. The project will be connected to bioinstrumentation, image analysis, computing and display devices, and biometric data readout systems. For Biomedical and Electrical Engineering, the student was taught the principles of electrical engineering and science as they apply to biotechnology and medicine. These are the skills needed for this project.

The remaining group members are in Computer Systems Engineering and Software Engineering, they will devote themselves to architecture designing and implementing for this project, they will also work in application design and implementation. The project will be based on Machine Learning. The architecture design will exhibition the basic structure of the project, then the system will be easier to understand. Furthermore, a good architecture design is not only benefiting the implementation at the beginning stage but also contributes to future development. Machine Learning has an essential position in the world, and we believe it will be more and more important. Studying machine learning will become an inevitable process and take this process earlier will make our career path easier.

Machine Learning is a study of computer algorithms that improve automatically through experience. Learning and using this technique will give extra more knowledge to students and will give students opportunity to train and combine the sills learned before.

Theoretically, image recognition is based on Deep Learning. Deep Learning, a subcategory of Machine Learning, refers to a set of automatic learning techniques and technologies based on artificial neural networks.

The techniques used in this project related to the courses SYSC 1005 Introduction to Software Development which includes the beginning of Python, SYSC 2100 Algorithms and Data Structures, SYSC 3110 Software Design Project, ELEC 3908 neural networks with MATLAB, and architecture design and measurable functional and non-functional requirements information learned from SYSC 3020 and SYSC 3120.

6.0 Projected progress

Timetable milestone

Date	Task	Deliverable
Fall 2021		
October 22, 2021	Work on Project proposal	Project proposal
October 29, 2021	Required skills in the project training	
November 5, 2021	Images collection	Required skills and knowledge
November 19, 2021	Machine learning library	
December 3, 2021	Start image processing	Image dataset
December 17, 2021	Feature selection, data analysis	Machine Learning Deep Learning techniques
Winter Break	Models build and train Models test and validate	Prototype Models
January 7, 2022	Working on Progress Report	Oral presentation form
January 16, 2022		Progress Report
January 21, 2022	Models update and correction	
January 28, 2022	Project update	
February 11, 2022	Test and Debug	Updated project models
February 25, 2022	Oral Presentation preparation	Final Report Draft
March 11, 2022	Working on Final Project Report	Oral Presentation
April 12, 2022		Final Project Report

7.0 Conclusion

In conclusion, our project is aimed to create a model involves a neural network that has equivalent or better judgements than a physician based on CT scan of the patient. The team has assessed challenges and risks. A strict timetable milestone will be kept professionally. Members of the team have a clear understanding of the project and the faith to overcome the difficulties by the next few months.

8.0 References

[1] Oden Tecchnologies. (n.d.). *What is Model Training?* [Online]. Available: https://oden.io/glossary/model-training/

[2] K. Vu. (2018, Nov. 29). *Beginner's Guide: Image Recognition and Deep Learning* [Online]. Available: https://dzone.com/articles/beginners-guide-image-recognition-and-deep-learnin

[3] Ipython. (n.d.). *The Jupyter Notebook* [Online]. Available: https://ipython.org/notebook.html

[4] J. Leban. (2020, May. 21). *Image recognition with Machine Learning on Python, Image processing* [Online]. Available: https://towardsdatascience.com/image-recognition-with-machine-learning-on-python-image-processing-3abe6b158e9a#79c6

[5] TensorFlow. (2021, Oct. 1). *Image classification* [Online]. Available: https://www.tensorflow.org/tutorials/images/classification#train_the_model